PATENT
Attorney Docket No.: SONY-27300

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:	Group Art Unit: 2621
Mikhail Dorojevets *et al.*	Examiner: Holder, Anner N
Serial No.: 10/816,391	**APPEAL BRIEF**
Filed: March 31, 2004	
For: **2D BLOCK PROCESSING ARCHITECTURE**	162 North Wolfe Road Sunnyvale, California 94086 (408) 530-9700
	Customer No. 28960

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In furtherance of the Applicants' Notice of Appeal filed on February 4, 2009, this Appeal Brief is submitted. This Appeal Brief is submitted in support of the Applicants' Notice of Appeal, and further pursuant to the rejection mailed on December 11, 2008, in which Claims 1-51 were rejected. The Applicants submit this Appeal Brief to the Board of Patent Appeals and Interferences in compliance with the requirements of 37 C.F.R. § 41.37, as stated in *Rules of Practice Before the Board of Patent Appeals and Interferences (Final Rule)*, 69 Fed. Reg. 49959 (August 12, 2004). The Applicants contend that the rejections of Claims 1-51 in this proceeding are in error, were previously overcome and are overcome again by this appeal.

I. REAL PARTIES IN INTEREST

As the assignee of the entire right, title, and interest in the above-captioned patent application, the real parties in interest in this appeal, is:

Sony Corporation, a Japanese corporation
6-7-35 Kitashinagawa, Shinagawa
Tokyo, 141
Japan

Sony Electronics Inc., a corporation of the State of Delaware
1 Sony Drive
Park Ridge, NJ 07656-8003

per the assignment document filed on August 13, 2004.

II. RELATED APPEALS AND INTERFERENCES

The Applicants are not aware of any other appeals or interferences related to the present application.

III. STATUS OF THE CLAIMS

Claims 1-51 are involved in the appeal. Claims 1, 2, 7, 8, 11, 13-16, 41, 44, 50 and 51 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over A Bit-Serial VLSI Array Processing Chip for Image Processing, IEEE Journal of Solid-State Circuits, Vol. 25, No. 2, April 1990 to Heaton et al. (hereinafter "Heaton," a copy of which is attached as Exhibit A). Claims 3, 4, 9, 10, 42, 43, 45 and 46 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of U.S. Patent No. 4,992,933 to Taylor (hereinafter "Taylor," a copy of which is attached as Exhibit B). Claims 12 and 49 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of U.S. Patent No. 4,745,547 to Buchholz (hereinafter "Buchholz," a copy of which is attached as Exhibit C). Claims 5, 6, 15, 47 and 48 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of U.S. Patent

No. 5,680,338 to Agarwal et al. (hereinafter "Agarwal," a copy of which is attached as Exhibit D). Claims 17-26, 28-38 and 40 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Taylor and further in view of Agarwal. Claims 27 and 39 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Taylor in view of Agarwal and further in view of Buchholz.

IV. STATUS OF THE AMENDMENTS FILED AFTER FINAL REJECTION

No amendments have been filed after the Office Action mailed on December 11, 2008.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The invention disclosed in the present application number 10/816,391 is directed to a video platform architecture for video processing includes complex video compression/decompression algorithms in a computer with a two-dimensional Single-Instruction Multiple-Data (SIMD) array architecture. The video platform architecture includes one or more video processing modules, on-chip shared memory, and a general-purpose RISC central processing unit CPU used as a system controller. Each video processing module includes a rectangular array of processing elements (PEs), a block load/store unit, a global-accumulation unit. Video to be processed is configured into blocks of data, and a general-purpose CPU used as a local controller. A plurality of registers are provided in the processing elements and the block load/store unit to support two-dimensional processing of the data blocks. Types of registers used include block registers, vector registers, scalar registers, and exchange registers. Each of these registers is designed to hold a short ordered one- or two-dimensional set of video data (data blocks). These registers are arranged in a hierarchical configuration along the data flow path between the on-chip memory and processing units within the PE array.

The elements of Claim 1, directed to one embodiment of the present invention, are described in the Specification at page 9, line 11 through page 13, line 13; page 15, line 24 through page 16, line 15 and the accompanying figures 1, 2 and 5. The video processing apparatus (10) comprises a memory (30) and one or more video processing modules (20), each video processing module (20) coupled to the memory (30) and comprising a programmable array of processing elements (100), each processing element including local registers to provide data used in processing operations and to store results of the processing operations, a block load and store unit (200) coupled to the programmable array of processing elements (100) to load, store, and send data transferred back and forth between the memory (30) and the array of processing

elements (100), a global accumulation unit (300) to accumulate the results of the processing operations for each processing element and a local controller (400) to provide instructions and parameters related to the processing operations and data transfer.

The elements of Claim 17, directed to one embodiment of the present invention, are described in the Specification at page 9, line 11 through page 13, line 13; page 16, line 16 through page 17, line 6 and the accompanying figures 1-3 and 6. The method comprises configuring a video stream into data blocks (600), loading data blocks from memory to a first array of exchange registers (602), loading data blocks from the first array of exchange registers to a programmable array of processing elements (604), wherein each processing element within the array of processing elements includes an array of block registers, an array of vector registers, and a local accumulator, the data blocks are loaded from the first array of exchange registers to the array of block registers, loading the data blocks from the array of block registers to the array of vector registers (606), processing the data blocks loaded in the array of vector registers and storing results in the corresponding local accumulator for each processing element (608), accumulating the results stored in the local accumulators in a global accumulator (612), thereby forming accumulated results and moving the accumulated results into a local controller.

The elements of Claim 29, directed to one embodiment of the present invention, are described in the Specification at page 9, line 11 through page 13, line 13; page 16, line 16 through page 17, line 6 and the accompanying figures 1-3 and 6. The video processing apparatus comprises means for configuring (50) a video stream into data blocks, means for loading (200) data blocks from memory to a first array of exchange registers, the means for loading (200) data blocks from memory coupled to the means for configuring, means for loading (200) data blocks from the first array of exchange registers to a programmable array of processing elements, the means for loading data blocks from the first array of exchange registers coupled to the means for loading data blocks from memory, wherein each processing element within the array of processing elements includes an array of block registers and an array of vector registers, the data blocks are loaded from the first array of exchange registers to the array of block registers, means for loading the data blocks from the array of block registers to the array of vector registers, the means for loading the data blocks from the array of block registers coupled to the means for loading data blocks from the first array of exchange registers, means for processing the data blocks loaded in the array of vector registers and storing results in the corresponding local accumulator for each processing element, the means for processing coupled to the means for loading the data blocks from the array of block registers, means for accumulating (300) the

results stored in the local accumulators in a global accumulator, thereby forming accumulated results, the means for accumulating coupled to the means for processing and means for moving the accumulated results into a local controller (400), the means for moving coupled to the means for accumulating.

A means for configuring a video stream into data blocks, referred to within the specification as a bit stream CPU 50 is shown in Figure 1. [Present Specification, page 9, line 14]

A means for loading data blocks from memory to a first array of exchange registers, referred to within the specification as a block load/store unit 200 is shown in Figure 2. [Present Specification, page 9, line 26 through page 10, line 17]

A means for loading data blocks from memory, referred to within the specification as a block load/store unit 200 is shown in Figure 2. [Present Specification, page 9, line 26 through page 10, line 17]

A means for loading the data blocks from the array of block registers to the array of vector registers, referred to within the specification as a block load/store unit 200 is shown in Figure 2. [Present Specification, page 9, line 26 through page 10, line 17]

A means for processing the data blocks loaded in the array of vector registers and storing results in the corresponding local accumulator for each processing element is shown as an ALU in Figure 3. [Present Specification, Figure 3]

A means for accumulating, referred to within the specification as a global accumulation unit 300 is shown in Figures 2 and 5. The global accumulation unit 300 is used to perform global accumulation of values. [Present Specification, page 15, line 24 through page 16, line 15]

A means for moving the accumulated results into a local controller, referred to within the specification as bus to a local CPU 400. The global accumulation result is compared by the local CPU 400. [Present Specification, page 17, lines 3-6]

The elements of Claim 41, directed to one embodiment of the present invention, are described in the Specification at page 9, line 11 through page 13, line 13; page 15, line 24 through page 16, line 15 and the accompanying figures 1-3 and 5. The programmable array of processing elements (100) to process video, each processing element comprises local registers to store video data blocks received from a main memory (30), to process the received video data blocks, and to store results of processing the video data blocks, wherein each processing element is configured to send the results to a global accumulation unit (300) to accumulate the results of the processing operations for each processing element.

VI. GROUNDS OF REJECTION AND OTHER MATTERS TO BE REVIEWED ON APPEAL

The following issues are presented in this Appeal Brief for review by the Board of Patent Appeals and Interferences:

1. Whether Claims 1, 2, 7, 8, 11, 13-16, 41, 44, 50 and 51 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton.
2. Whether Claims 3, 4, 9, 10, 42, 43, 45 and 46 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Taylor.
3. Whether Claims 12 and 49 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Buchholz.
4. Whether Claims 5, 6, 15, 47 and 48 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Agarwal.
5. Whether Claims 17-26, 28-38 and 40 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Taylor and further in view of Agarwal.
6. Whether Claims 27 and 39 are properly rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Taylor in view of Agarwal and further in view of Buchholz.

VII. ARGUMENT

Grounds for Rejection

Within the Office Action, Claims 1, 2, 7, 8, 11, 13-16, 41, 44, 50 and 51 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton.

Outline of Arguments

In the discussion that follows, the Applicants discuss the teachings of Heaton. As will be discussed in detail below, Heaton does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

1. Heaton does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

Heaton teaches an array processing chip which integrates many processing elements on a single die. Each processing element has several components including a 16-function logical unit, an adder, a shift register and local RAM. [Heaton, Abstract] The shift register is used as a local accumulator to hold arithmetic operands. [Heaton, page 366, ¶2] Heaton also teaches:

> An OR tree is connected to all PE's on the chip, enabling the values presented on each of the 128 PE data buses to be ORed together. This feature enables the user to quickly test for a "true" bit in any of the PE's of the array. The OR tree is useful in associative operations and in performing data searches. OR tree operations are pipelined. The single OR pin output is open drain, enabling several BLITZEN chips to be directly wire ORed together. [Heaton, page 367, ¶2]

The Office Action also cites Figure 6 of Heaton. Figure 6 of Heaton shows a processing element with a final output to a SUM-OR tree. There is no hint, teaching or suggestion that a global accumulator is implemented. Further, the Introduction of Heaton is also cited within the Office Action. However, the Introduction of Heaton merely describes parallel processing systems in general and provides no hint, teaching or suggestion regarding a global accumulator. Thus, Heaton does not teach, hint or suggest a global accumulation unit to accumulate the results of the processing operations for each processing element.

Furthermore, the Office Action does not provide a justification or motivation as to why the local accumulator of Heaton is able to be transformed into a global accumulator to reject the claimed invention. Heaton clearly only teaches a local accumulator and does not teach, hint or suggest a global accumulator. Since Heaton only teaches a local accumulator, it is clearly improper to extrapolate from a local accumulator that a global accumulator is obvious. It is only with the knowledge of the claimed invention, and using improper hindsight, that the rejection is able to be made. Using the knowledge of claimed invention to reject itself is clearly impermissible.

In contrast to Heaton, the presently claimed invention is directed to a video platform architecture for video processing which includes complex video compression/decompression algorithms in a computer with a two-dimensional Single-Instruction Multiple-Data (SIMD) array architecture. The video platform architecture includes one or more video processing modules, on-chip shared memory, and a general-purpose RISC central processing unit CPU used as a

system controller. Each video processing module includes a rectangular array of processing elements (PEs), a block load/store unit and a global-accumulation unit. Video to be processed is configured into blocks of data and a general-purpose CPU used as a local controller. A plurality of registers are provided in the processing elements and the block load/store unit to support two-dimensional processing of the data blocks. Types of registers used include block registers, vector registers, scalar registers, and exchange registers. Each of these registers is designed to hold a short ordered one- or two-dimensional set of video data (data blocks). These registers are arranged in a hierarchical configuration along the data flow path between the on-chip memory and processing units within the PE array. [Present Specification, Abstract]

Furthermore, in some embodiments, the global accumulation unit includes 4 slice accumulation (SACC) registers, 1 global PE mask control register, and 1 global accumulation (GACC) register. In some embodiments, there is one SACC register for each vertical PE slice of the PE array. The SACC registers are the intermediate registers in the operations moving data from the LACC register of each PE to the GACC register. In some embodiments, there are 4 40-bit SACC registers in the global accumulation unit. Each of the SACC registers includes three individually written sections, namely low 16-bits, middle 16-bits, and high 8-bits. Each PE's 40-bit LACC is read in steps, specifying which part of the LACC, low 16-bits, middle 16-bits, or high 8-bits, is to be placed on the 16-bit bus to the global accumulation unit, and finally into corresponding section of the appropriate SACC register. During operation of the global accumulation unit, either the full 40-bit values or packed 20-bit values of the SACC register involved in the accumulation operations are added together by a global add instruction and a global add and accumulate instruction. The GACC register is used to perform global accumulation of LACC values from multiple PEs loaded into the corresponding SACC registers. In some embodiments, there is one 48-bit GACC register in the global accumulation unit. [Present Specification, page 15, line 24 through page 16, line 8] As described above, Heaton does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

It is asserted within the Response to Arguments section of the Office Action that since Heaton teaches local accumulation, it would have been obvious and is fairly suggested by the reference to store results for each processing element in one central location (global accumulation unit). The Applicants respectfully disagree with this assertion. The Examiner's assertion is based on improper hindsight reasoning. Heaton does not suggest, teach or disclose a global accumulation unit as claimed. Instead, Heaton teaches a local accumulation unit that is a

two operand adder. Heaton also teaches an OR tree. However, a global accumulation unit, as suggested by the Examiner, would then require an additional adder and a shift register, similar to those used in Heaton's local accumulation unit, as described on page 366, ¶2 and Figure 6. Such a global accumulation unit, however, would not be able to accumulate the results of the precessing operations for each processing element, as taught in the present specification; rather, this global accumulation unit would only add two results together. Further, Heaton's OR tree is connected to all PE's on the chip, enabling the values presented on each of the 128 data buses to be ORed together. OR tree operations are pipelined. As such, the OR tree performs a logical OR operation, not an accumulation operation. Accordingly, the Applicants respectfully submit that Heaton does not suggest, teach or disclose a global accumulation unit as claimed.

2. The claims distinguish over Heaton.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claims 1, 2, 7, 8, 11, 13-16

The independent Claim 1 is directed to a video processing apparatus. The video processing apparatus of Claim 1 comprises a memory and one or more video processing modules, each video processing module coupled to the memory and comprising a programmable array of processing elements, each processing element including local registers to provide data used in processing operations and to store results of the processing operations, a block load and store unit coupled to the programmable array of processing elements to load, store, and send data transferred back and forth between the memory and the array of processing elements, a global accumulation unit to accumulate the results of the processing operations for each processing element and a local controller to provide instructions and parameters related to the processing operations and data transfer. As described above, Heaton does not teach or make obvious a global accumulation unit to accumulate the results of the processing operations for each processing element. For at least these reasons, the independent Claim 1 is allowable over the teachings of Heaton.

Claims 2, 7, 8, 11 and 13-16 are all dependent upon the independent Claim 1. As described above, the independent Claim 1 is allowable over the teachings of Heaton.

Accordingly, Claims 2, 7, 8, 11 and 13-16 are all also allowable as being dependent upon an allowable base claim.

b. Claims 41, 44, 50 and 51

The independent Claim 41 is directed to a programmable array of processing elements to process video, each processing element comprising local registers to store video data blocks received from a main memory, to process the received video data blocks, and to store results of processing the video data blocks, wherein each processing element is configured to send the results to a global accumulation unit to accumulate the results of the processing operations for each processing element. As described above, Heaton does not teach or make obvious wherein each processing element is configured to send the results to a global accumulation unit to accumulate the results of the processing operations for each processing element. For at least these reasons, the independent Claim 41 is allowable over the teachings of Heaton.

Claims 44, 50 and 51 are all dependent upon the independent Claim 41. As described above, the independent Claim 41 is allowable over the teachings of Heaton. Accordingly, Claims 44, 50 and 51 are all also allowable as being dependent upon an allowable base claim.

Grounds for Rejection

Within the Office Action, Claims 3, 4, 9, 10, 42, 43, 45 and 46 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Taylor.

The claims distinguish over Heaton, Taylor and their combination.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claims 3, 4, 9 and 10

Claims 3, 4, 9 and 10 are all dependent upon the independent Claim 1. As described above, the independent Claim 1 is allowable over the teachings of Heaton. Accordingly, Claims 3, 4, 9 and 10 are all also allowable as being dependent upon an allowable base claim.

b. Claims 42, 43, 45 and 46

Claims 42, 43, 45 and 46 are all dependent upon the independent Claim 41. As described above, the independent Claim 41 is allowable over the teachings of Heaton. Accordingly, Claims 42, 43, 45 and 46 are all also allowable as being dependent upon an allowable base claim.

Grounds for Rejection

Within the Office Action, Claims 12 and 49 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Buchholz.

The claims distinguish over Heaton, Buchholz and their combination.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claim 12

Claim 12 is dependent upon the independent Claim 1. As described above, the independent Claim 1 is allowable over the teachings of Heaton. Accordingly, Claim 12 is also allowable as being dependent upon an allowable base claim.

b. Claim 49

Claim 49 is dependent upon the independent Claim 41. As described above, the independent Claim 41 is allowable over the teachings of Heaton. Accordingly, Claim 49 is also allowable as being dependent upon an allowable base claim.

Grounds for Rejection

Within the Office Action, Claims 5, 6, 15, 47 and 48 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Agarwal.

The claims distinguish over Heaton, Agarwal and their combination.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claims 5, 6, 15

Claims 5, 6 and 15 are all dependent upon the independent Claim 1. As described above, the independent Claim 1 is allowable over the teachings of Heaton. Accordingly, Claims 5, 6 and 15 are all also allowable as being dependent upon an allowable base claim.

b. Claims 47 and 48

Claims 47 and 48 are all dependent upon the independent Claim 41. As described above, the independent Claim 41 is allowable over the teachings of Heaton. Accordingly, Claims 47 and 48 are all also allowable as being dependent upon an allowable base claim.

Grounds for Rejection

Within the Office Action, Claims 17-26, 28-38 and 40 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Taylor and further in view of Agarwal.

Outline of Arguments

In the discussion that follows, the Applicants discuss the teachings of Heaton, the teachings of Taylor, the teachings of Agarwal and the teachings of their combination. As will be discussed in detail below, Heaton, Taylor, Agarwal and their combination do not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

1. Heaton does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element. It is recognized that Heaton does not teach moving results into a local controller or loading data blocks from the array of block registers to the array of vector registers.

Heaton teaches an array processing chip which integrates many processing elements on a single die. Each processing element has several components including a 16-function logical unit, an adder, a shift register and local RAM. [Heaton, Abstract] The shift register is used as a local accumulator to hold arithmetic operands. [Heaton, page 366, ¶2] Heaton also teaches:

> An OR tree is connected to all PE's on the chip, enabling the values presented on each of the 128 PE data buses to be ORed together. This feature enables the user to quickly test for a "true" bit in any of the PE's of the array. The OR tree is useful in associative operations and in performing data searches. OR tree operations are pipelined. The single OR pin output is open drain, enabling several BLITZEN chips to be directly wire ORed together. [Heaton, page 367, ¶2]

The Office Action also cites Figure 6 of Heaton. Figure 6 of Heaton shows a processing element with a final output to a SUM-OR tree. There is no hint, teaching or suggestion that a global accumulator is implemented. Further, the Introduction of Heaton is also cited within the Office Action. However, the Introduction of Heaton merely describes parallel processing systems in general and provides no hint, teaching or suggestion regarding a global accumulator. Thus, Heaton does not teach, hint or suggest a global accumulation unit to accumulate the results of the processing operations for each processing element.

Furthermore, the Office Action does not provide a justification or motivation as to why the local accumulator of Heaton is able to be transformed into a global accumulator to reject the claimed invention. Heaton clearly only teaches a local accumulator and does not teach, hint or suggest a global accumulator. Since Heaton only teaches a local accumulator, it is clearly improper to extrapolate from a local accumulator that a global accumulator is obvious. It is only with the knowledge of the claimed invention, and using improper hindsight, that the rejection is able to be made. Using the knowledge of claimed invention to reject itself is clearly impermissible.

In contrast to Heaton, the presently claimed invention is directed to a video platform architecture for video processing which includes complex video compression/decompression algorithms in a computer with a two-dimensional Single-Instruction Multiple-Data (SIMD) array

architecture. The video platform architecture includes one or more video processing modules, on-chip shared memory, and a general-purpose RISC central processing unit CPU used as a system controller. Each video processing module includes a rectangular array of processing elements (PEs), a block load/store unit and a global-accumulation unit. Video to be processed is configured into blocks of data and a general-purpose CPU used as a local controller. A plurality of registers are provided in the processing elements and the block load/store unit to support two-dimensional processing of the data blocks. Types of registers used include block registers, vector registers, scalar registers, and exchange registers. Each of these registers is designed to hold a short ordered one- or two-dimensional set of video data (data blocks). These registers are arranged in a hierarchical configuration along the data flow path between the on-chip memory and processing units within the PE array. [Present Specification, Abstract]

Furthermore, in some embodiments, the global accumulation unit includes 4 slice accumulation (SACC) registers, 1 global PE mask control register, and 1 global accumulation (GACC) register. In some embodiments, there is one SACC register for each vertical PE slice of the PE array. The SACC registers are the intermediate registers in the operations moving data from the LACC register of each PE to the GACC register. In some embodiments, there are 4 40-bit SACC registers in the global accumulation unit. Each of the SACC registers includes three individually written sections, namely low 16-bits, middle 16-bits, and high 8-bits. Each PE's 40-bit LACC is read in steps, specifying which part of the LACC, low 16-bits, middle 16-bits, or high 8-bits, is to be placed on the 16-bit bus to the global accumulation unit, and finally into corresponding section of the appropriate SACC register. During operation of the global accumulation unit, either the full 40-bit values or packed 20-bit values of the SACC register involved in the accumulation operations are added together by a global add instruction and a global add and accumulate instruction. The GACC register is used to perform global accumulation of LACC values from multiple PEs loaded into the corresponding SACC registers. In some embodiments, there is one 48-bit GACC register in the global accumulation unit. [Present Specification, page 15, line 24 through page 16, line 8] As described above, Heaton does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

It is asserted within the Response to Arguments section of the Office Action that since Heaton teaches local accumulation, it would have been obvious and is fairly suggested by the reference to store results for each processing element in one central location (global accumulation unit). The Applicants respectfully disagree with this assertion. The Examiner's

assertion is based on improper hindsight reasoning. Heaton does not suggest, teach or disclose a global accumulation unit as claimed. Instead, Heaton teaches a local accumulation unit that is a two operand adder. Heaton also teaches an OR tree. However, a global accumulation unit, as suggested by the Examiner, would then require an additional adder and a shift register, similar to those used in Heaton's local accumulation unit, as described on page 366, ¶2 and Figure 6. Such a global accumulation unit, however, would not be able to accumulate the results of the precessing operations for each processing element, as taught in the present specification; rather, this global accumulation unit would only add two results together. Further, Heaton's OR tree is connected to all PE's on the chip, enabling the values presented on each of the 128 data buses to be ORed together. OR tree operations are pipelined. As such, the OR tree performs a logical OR operation, not an accumulation operation. Accordingly, the Applicants respectfully submit that Heaton does not suggest, teach or disclose a global accumulation unit as claimed.

2. Taylor does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

Taylor teaches a single-instruction-multiple-data (SIMD) array processor with a multi-dimensional array of processing elements and control logic for issuing global instructions to the array. Taylor also teaches that each processing element in the array has individually programmable instruction decoder and a mechanism which enables efficiently programming and reprogramming of the instruction decoder. [Taylor, Abstract] However, Taylor does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

3. Agarwal does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

Agarwal teaches a vector processing system for processing vector calculations utilizing a portion of a vector comprising a plurality of elements, means for receiving a vector and a vector processing command are provided. Agarwal also teaches the vector processing system includes means for receiving and storing a start-element value and an end-element value. Agarwal teaches an arithmetic logic unit is coupled to the means for receiving the vector, the means for receiving the vector processing command, and the means for receiving the start-element and end-element

values. [Agarwal, Abstract] However, Agarwal does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

4. The combination of Heaton, Taylor and Agarwal does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

As described above, Heaton does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element. Taylor and Agarwal are apparently cited as teaching moving results into a local controller and loading data blocks from the array of block registers to the array of vector registers. However, Taylor and Agarwal also do not teach a global accumulation unit to accumulate the results of the processing operations for each processing element. Thus, the combination of Heaton, Taylor and Agarwal does not teach a global accumulation unit to accumulate the results of the processing operations for each processing element.

5. The claims distinguish over Heaton, Taylor, Agarwal and their combination.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claims 17-26 and 28

The independent Claim 17 is directed to a method of processing video. The method of Claim 17 comprises configuring a video stream into data blocks, loading data blocks from memory to a first array of exchange registers, loading data blocks from the first array of exchange registers to a programmable array of processing elements, wherein each processing element within the array of processing elements includes an array of block registers, an array of vector registers, and a local accumulator, the data blocks are loaded from the first array of exchange registers to the array of block registers, loading the data blocks from the array of block registers to the array of vector registers, processing the data blocks loaded in the array of vector registers and storing results in the corresponding local accumulator for each processing element, accumulating the results stored in the local accumulators in a global accumulator, thereby

forming accumulated results and moving the accumulated results into a local controller. As described above, neither Heaton, Taylor, Agarwal nor their combination teach accumulating the results stored in the local accumulators in a global accumulator, thereby forming accumulated results. For at least these reasons, the independent Claim 17 is allowable over the teachings of Heaton, Taylor, Agarwal and their combination.

Claims 18-26 and 28 are all dependent upon the independent Claim 17. As described above, the independent Claim 17 is allowable over the teachings of Heaton, Taylor, Agarwal and their combination. Accordingly, Claims 18-26 and 28 are all also allowable as being dependent upon an allowable base claim.

b. Claims 29-38 and 40

The independent Claim 29 is directed to a video processing apparatus. The video processing apparatus of Claim 29 comprises means for configuring a video stream into data blocks, means for loading data blocks from memory to a first array of exchange registers, the means for loading data blocks from memory coupled to the means for configuring, means for loading data blocks from the first array of exchange registers to a programmable array of processing elements, the means for loading data blocks from the first array of exchange registers coupled to the means for loading data blocks from memory, wherein each processing element within the array of processing elements includes an array of block registers and an array of vector registers, the data blocks are loaded from the first array of exchange registers to the array of block registers, means for loading the data blocks from the array of block registers to the array of vector registers, the means for loading the data blocks from the array of block registers coupled to the means for loading data blocks from the first array of exchange registers, means for processing the data blocks loaded in the array of vector registers and storing results in the corresponding local accumulator for each processing element, the means for processing coupled to the means for loading the data blocks from the array of block registers, means for accumulating the results stored in the local accumulators in a global accumulator, thereby forming accumulated results, the means for accumulating coupled to the means for processing and means for moving the accumulated results into a local controller, the means for moving coupled to the means for accumulating. As described above, neither Heaton, Taylor, Agarwal nor their combination teach means for accumulating the results stored in the local accumulators in a global accumulator, thereby forming accumulated results. For at least these reasons, the

independent Claim 29 is allowable over the teachings of Heaton, Taylor, Agarwal and their combination.

Claims 30-38 and 40 are all dependent upon the independent Claim 29. As described above, the independent Claim 29 is allowable over the teachings of Heaton, Taylor, Agarwal and their combination. Accordingly, Claims 30-38 and 40 are all also allowable as being dependent upon an allowable base claim.

Grounds for Rejection

Within the Office Action, Claims 27 and 39 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Heaton in view of Taylor in view of Agarwal and further in view of Buchholz.

The claims distinguish over Heaton, Taylor, Agarwal, Buchholz and their combination.

The claims are grouped separately below to indicate that they do not stand or fall together.

a. Claim 27

Claim 27 is dependent upon the independent Claim 17. As described above, the independent Claim 17 is allowable over the teachings of Heaton, Taylor, Agarwal and their combination. Accordingly, Claim 27 is also allowable as being dependent upon an allowable base claim.

b. Claim 39

Claim 39 is dependent upon the independent Claim 29. As described above, the independent Claim 29 is allowable over the teachings of Heaton. Accordingly, Claim 39 is also allowable as being dependent upon an allowable base claim.

CONCLUSION

For the above reasons, it is respectfully submitted that the Claims 1-51 are allowable over the cited prior art references. Therefore, a favorable indication is respectfully requested.

Respectfully submitted,
HAVERSTOCK & OWENS LLP

Dated: March 20, 2009

By: /Jonathan O. Owens/
Jonathan O. Owens
Reg. No.: 37,902
Attorney for Applicant

VIII. CLAIMS APPENDIX

This appendix includes a list of the claims under appeal.

1. (original) A video processing apparatus comprising:
   a. a memory; and
   b. one or more video processing modules, each video processing module coupled to the memory and comprising:
      i. a programmable array of processing elements, each processing element including local registers to provide data used in processing operations and to store results of the processing operations;
      ii. a block load and store unit coupled to the programmable array of processing elements to load, store, and send data transferred back and forth between the memory and the array of processing elements;
      iii. a global accumulation unit to accumulate the results of the processing operations for each processing element; and
      iv. a local controller to provide instructions and parameters related to the processing operations and data transfer.

2. (original) The apparatus of claim 1 wherein the array of processing elements comprises a two-dimensional array.

3. (original) The apparatus of claim 2 wherein the two-dimensional array comprises a 4x4 array of processing elements.

4. (original) The apparatus of claim 2 wherein the two-dimensional array comprises a single-instruction multiple-data array.

5. (original) The apparatus of claim 1 wherein each processing element includes a plurality of vector registers and a plurality of block registers.

6. (original) The apparatus of claim 5 wherein each vector register and each block register is configured to hold 8 8-bit data elements as a two-dimensional 2x4 block of pixels or 4 16-bit data elements as a one-dimensional vector.

7. (original) The apparatus of claim 1 wherein the block load and store unit comprises one or more arrays of exchange registers.

8. (original) The apparatus of claim 7 wherein each array of exchange registers is a two-dimensional array.

9. (original) The apparatus of claim 1 wherein the local controller provides control commands to each processing element, performing control and processing operations on data stored within the local controller, and transfers data between the local controller and other registers within one video module.

10. (original) The apparatus of claim 1 further comprising a system controller coupled to the memory and to the one or more video processing modules.

11. (original) The apparatus of claim 1 further comprising a direct, high-bandwidth data path to couple each of the video processing modules to the memory.

12. (original) The apparatus of claim 1 wherein each processing element further comprises a plurality of scalar registers.

13. (original) The apparatus of claim 1 wherein the block load and store unit sends data transferred back and forth between non-adjacent processing elements of the array of processing elements.

14. (original) The apparatus of claim 1 wherein each processing element includes a local accumulation register.

15. (original) The apparatus of claim 1 wherein each processing element further comprises a plurality of control registers including a PE mask register, a condition register, a block base register, and a vector base register.

16. (original) The apparatus of claim 1 wherein the block load and store unit sends data transferred back and forth between the local registers in the processing elements, the global accumulation unit, and the local controller.

17. (original) A method of processing video comprising:
    a. configuring a video stream into data blocks;
    b. loading data blocks from memory to a first array of exchange registers;
    c. loading data blocks from the first array of exchange registers to a programmable array of processing elements, wherein each processing element within the array of processing elements includes an array of block registers, an array of vector registers, and a local accumulator, the data blocks are loaded from the first array of exchange registers to the array of block registers;
    d. loading the data blocks from the array of block registers to the array of vector registers;
    e. processing the data blocks loaded in the array of vector registers and storing results in the corresponding local accumulator for each processing element;
    f. accumulating the results stored in the local accumulators in a global accumulator, thereby forming accumulated results; and
    g. moving the accumulated results into a local controller.

18. (original) The method of claim 17 further comprising storing results from processing the data blocks in the array of vector registers, and loading the results stored in the array of vector registers in the array of block registers.

19. (original) The method of claim 18 further comprising loading the results in the array of block registers into a second array of exchange registers, and loading the results from the array of block registers into memory.

20. (original) The method of claim 19 wherein each of the first and second array of exchange registers is a two-dimensional array.

21. (original) The method of claim 18 further comprising loading the results in the array of block registers into a second array of exchange registers, and loading the results in the second array of exchange registers into another array of block registers included within non-adjacent processing elements to the processing elements including the array of block registers.

22. (original) The method of claim 18 further comprising loading the results in the array of block registers into another array of block registers included within a processing element adjacent to the processing element including the array of block registers.

23. (original) The method of claim 17 wherein the array of processing elements comprises a two-dimensional array.

24. (original) The method of claim 23 wherein the two-dimensional array comprises a 4x4 array of processing elements.

25. (original) The method of claim 23 wherein the two-dimensional array comprises a single-instruction multiple-data array.

26. (original) The method of claim 17 wherein each vector register and each block register is configured to hold 8 8-bit data elements as a two-dimensional 2x4 block of pixels or 4 16-bit data elements as a one-dimensional vector.

27. (original) The method of claim 17 wherein each processing element further comprises a plurality of scalar registers such that processing the data blocks includes processing data blocks loaded from the array of block registers and data loaded from the array of scalar registers.

28. (original) The method of claim 17 wherein the local controller utilizes the accumulated results to make control decisions related to video processing.

29. (original) A video processing apparatus comprising:

a. means for configuring a video stream into data blocks;

b. means for loading data blocks from memory to a first array of exchange registers, the means for loading data blocks from memory coupled to the means for configuring;

c. means for loading data blocks from the first array of exchange registers to a programmable array of processing elements, the means for loading data blocks from the first array of exchange registers coupled to the means for loading data blocks from memory, wherein each processing element within the array of processing elements includes an array of block registers and an array of vector registers, the data blocks are loaded from the first array of exchange registers to the array of block registers;

d. means for loading the data blocks from the array of block registers to the array of vector registers, the means for loading the data blocks from the array of block registers coupled to the means for loading data blocks from the first array of exchange registers;

e. means for processing the data blocks loaded in the array of vector registers and storing results in the corresponding local accumulator for each processing element, the means for processing coupled to the means for loading the data blocks from the array of block registers;

f. means for accumulating the results stored in the local accumulators in a global accumulator, thereby forming accumulated results, the means for accumulating coupled to the means for processing; and

g. means for moving the accumulated results into a local controller, the means for moving coupled to the means for accumulating.

30. (original) The apparatus of claim 29 further comprising means for storing results from processing the data blocks in the array of vector registers, and means for loading the results stored in the array of vector registers in the array of block registers.

31. (original) The apparatus of claim 30 further comprising means for loading the results in the array of block registers into a second array of exchange registers, and means for loading the results from the array of block registers into memory.

32. (original) The apparatus of claim 31 wherein each of the first and second array of exchange registers is a two-dimensional array.

33. (original) The apparatus of claim 30 further comprising means for loading the results in the array of block registers into a second array of exchange registers, and means for loading the results in the second array of exchange registers into another array of block registers included within non-adjacent processing elements to the processing elements including the array of block registers.

34. (original) The apparatus of claim 30 further comprising means for loading the results in the array of block registers into another array of block registers included within a processing element adjacent to the processing element including the array of block registers.

35. (original) The apparatus of claim 29 wherein the array of processing elements comprises a two-dimensional array.

36. (original) The apparatus of claim 35 wherein the two-dimensional array comprises a 4x4 array of processing elements.

37. (original) The apparatus of claim 35 wherein the two-dimensional array comprises a single-instruction multiple-data array.

38. (original) The apparatus of claim 29 wherein each vector register and each block register is configured to hold 8 8-bit data elements as a two-dimensional 2x4 block of pixels or 4 16-bit data elements as a one-dimensional vector.

39. (original) The apparatus of claim 29 wherein each processing element further comprises a plurality of scalar registers such that processing the data blocks includes processing data

blocks loaded from the array of block registers and data loaded from the array of scalar registers.

40. (original) The apparatus of claim 29 wherein the local controller utilizes the accumulated results to make control decisions related to video processing.

41. (previously presented) A programmable array of processing elements to process video, each processing element comprising:

local registers to store video data blocks received from a main memory, to process the received video data blocks, and to store results of processing the video data blocks, wherein each processing element is configured to send the results to a global accumulation unit to accumulate the results of the processing operations for each processing element.

42. (original) The programmable array of processing elements of claim 41 coupled to a local controller to provide instructions and parameters related to data transfer and processing of the video data blocks received from the main memory.

43. (original) The programmable array of processing elements of claim 42 wherein the local controller provides control commands to each processing element, performing control and processing operations on data stored within the local controller, and transfers data between the local controller and other registers within one video module.

44. (original) The programmable array of processing elements of claim 41 wherein the array of processing elements comprises a two-dimensional array.

45. (original) The programmable array of processing elements of claim 44 wherein the two-dimensional array comprises a 4x4 array of processing elements.

46. (original) The programmable array of processing elements of claim 44 wherein the two-dimensional array comprises a single-instruction multiple-data array.

47. (original) The programmable array of processing elements of claim 41 wherein each processing element includes a plurality of vector registers and a plurality of block registers.

48. (original) The programmable array of processing elements of claim 47 wherein each vector register and each block register is configured to hold 8 8-bit data elements as a two-dimensional 2x4 block of pixels or 4 16-bit data elements as a one-dimensional vector

49. (original) The programmable array of processing elements of claim 41 wherein each processing element further comprises a plurality of scalar registers.

50. (original) The programmable array of processing elements of claim 41 wherein each processing element includes a local accumulation register.

51. (original) The programmable array of processing elements of claim 41 wherein each processing element further comprises a plurality of control registers including a PE mask register, a condition register, a block base register, and a vector base register.

IX. EVIDENCE APPENDIX

STATEMENT

Pursuant to 37 C.F.R. § 41.37(c)(1)(ix), the following is a statement setting forth where in the record the evidence of this appendix was entered by the examiner:

Evidence Description:	**Where Entered:**
A Bit-Serial VLSI Array Processing Chip for Image Processing, IEEE Journal of Solid-State Circuits, Vol. 25, No. 2, April 1990 to Heaton et al.	Information Disclosure Statement filed April 13, 2006 and considered December 11, 2007
U.S. Pat. No. 4,992,933	Office Action mailed January 28, 2008
U.S. Pat. No. 4,745,547	Office Action mailed January 28, 2008
U.S. Pat. No. 5,680,338	Office Action mailed January 28, 2008
Office Action December 11, 2008	Examiner Office Action

X. RELATED PROCEEDINGS APPENDIX

There are no related proceedings.